|  |
| --- |
| Text  Description automatically generated  **Submitted by:**  **Name: Hasan Tanveer Mahmood**  **Matric no: 1725413**  **COMPUTER ARCHITECTURE & ASSEMBLY LANGUAGE**  **Course: CSC-3402, Sec: 02**  **Lecturer: Dr. HAFIZAH BINTI MANSOR** A picture containing text  Description automatically generated **LAB EXERCISE - 1** |

**DATE: 25-MAR-2021**

**Lab 1 Questions:**

Using MARS simulator to simulate and debug your source code, create a program (using MIPS instruction set) to have the following features:

1. Print out a message string with a welcome message "Welcome to Lab 1 exercise. Please enter your name "

2. Get an input from user to enter his/her name

3. Print out a message string with a hello message and followed by the user's name (e.g. "Hello Adam")

4. Manipulate the user's name to output an encrypted user name with a 7 place displacement (i.e. A=H, d=k, a=h, m=t)

5. Print out the encrypted user name with a short message ("Hi Adam, your encrypted user name is Hkht")

6. End the code

7. Optimise the code using procedures whenever possible.

**Source Code:**

|  |
| --- |
| # Name: Hasan Tanveer Mahmood  # Matric no: 1725413  # CSC 3402, Sec: 2  .data    WelcomeMsg: .asciiz "Welcome to Lab 1 exercise. Please Enter Your Name: "  HelloMsg: .asciiz "Hi, "  name: .space 10  EncryptMsg: .asciiz "your encrypted user name is : "    .text  main:    # Print out a message string with a welcome message.  li $v0,4  la $a0, WelcomeMsg  syscall    # Get an input from user to enter his/her name  la $a0,name  la $a1,10  li $v0,8  syscall  la $t0,($a0) # stored name in the register t0  li $t1,0 # string length    # Print out a message string with a hello message and followed by the user's name  li $v0, 4 # load immediate and prepare to print.  la $a0, HelloMsg # load address of HelloMsg  syscall  li $v0, 4  la $a0, name  syscall    # Print out the encrypted user name with a short message  li $v0, 4  la $a0, EncryptMsg  syscall  # Encryption  Encrypt:  lb $t4, 0($t0) # The first character of the name is read  beq $t6,10,end # Terminate program on the \n  beqz $t4,end # Terminate Program when the end of the string is reached  jal islower    # Check if the character is lower case  Encrypt2:  beq $v0,1,EncryptLower  beq $v0,0,EncryptUpper  move $a0, $t4    # Function for printing Encrypted charecter  PrintEncryptChar:  li $v0,11 # load immidiate for printing the encrypted charecter  syscall  add $t0,$t0,1 # Point to the next charecter  add $t1,$t1,1  j Encrypt      # End the code  end:  li $v0,10  syscall    # Funtion for encrypting the name's charecter.  islower:  bgt $t4,122,NotlowerOrupper  blt $t4,97,CheckCase  li $v0,1  jr $ra # return back to the return address  NotlowerOrupper:  li $v0,2 # store value 2 in register VO if the character is not lower or upper  j Encrypt2 # move back    # Here program will check if the case of charecter is upper or any other  CheckCase:  blt $t4,65,NotlowerOrupper # if the character is not upper or lower  bgt $t4,91,NotlowerOrupper # if character is not upper or lower  li $v0,0 # store value 0 in register V0 if the character is upper case  j Encrypt2    # Manipulate the user's name lowercases to output an encrypted user name with a 7 place displacement  EncryptLower:  li $t5,26  sub $t4,$t4,97  add $t4, $t4, 7  div $t4,$t5  mfhi $a0  addi $a0,$a0,97    j PrintEncryptChar    # Manipulate the user's name uppercases to output an encrypted user name with a 7 place displacement  EncryptUpper:  li $t5,26  sub $t4,$t4,65  add $t4, $t4, 7  div $t4,$t5  mfhi $a0  addi $a0,$a0,65  j PrintEncryptChar |

**Graphical user interface, text

Description automatically generatedScreen Capture of the Out Put:**